

# Linux et logiciels libres en réseau

*L'atelier est animé par Bernard LANG, Secrétaire général, AFUL.*

*Intervenants :*

*Philippe ALLART, Illico*

*Denis DUCAMP, HSC*

*Stéphane FERMIGIER, Nuxeo*

*Jean-Pierre LAISNE, Linbox*

*Jean-Paul SMETS, DRIRE Lorraine*

*Serdjane TIJANIC, IdealX*

## Les logiciels libres sur les réseaux

**Bernard LANG**

**Secrétaire général, AFUL**

Cet atelier a pour ambition de dresser un panorama de l'utilisation de Linux et des logiciels libres en réseau. En effet, il s'agit d'une des plates-formes les plus importantes sur le réseau. On oublie trop souvent que Linux est le système d'exploitation dominant sur le marché mondial des serveurs Internet. Le marché des serveurs Internet est dominé par le logiciel Apache qui tournait principalement sur des plates-formes Unix et qui représentait une part de marché de 62 % en nombre d'installations (la valorisation en dollars n'est pas significative pour les logiciels libres). En dessous du logiciel serveur, plusieurs systèmes d'exploitation (OS) sont en concurrence : Linux représente 32 % du marché, Windows NT 26 % et FreeBSD 20 %. Sur intranet, les chiffres sont probablement différents, mais les éléments statistiques ne sont pas disponibles. Ces statistiques sont issues d'une enquête de la société Netcraft qui sonde plusieurs centaines de milliers de serveurs.

Les logiciels libres ne sont donc pas des outils marginaux dans ce secteur. D'une manière générale Linux est utilisée pour satisfaire les besoins Unix habituels et les besoins d'utilisateurs Windows. Dans ce domaine, la part de marché de Linux a crû entre 1999 et 2000 de 25 % à 27 %.

# La messagerie unifiée

**Serdjane TIJANIC**  
**IDEALX**

## **I. La conception de IDX-MU**

### **1. Nos objectifs**

En réponse à des solutions propriétaires comme Microsoft Exchange, Nous avons lancé IDX-MU. Nous visons à :

- harmoniser les messageries autour de standards Internet ;
- apporter une fiabilité et une robustesse ;
- garantir la sécurité ;
- réduire le coût total de possession (TCO).

Lors de la conception de ce produit, nous avons identifié les besoins de nos clients. Il est apparu que ces derniers recherchaient :

- un service disponible ;`
- un système au niveau du savoir-faire actuel ;
- une gestion centralisée et maîtrisable
- un haut niveau de sécurité.

### **2. Procédure**

Nous nous sommes appuyés sur

- des briques Open Source éprouvées à l'échelle mondiale ;
- une architecture modulaire dérivée du monde Open Source et d'Internet en agencant des piles de base ;
- un annuaire LDAP qui centralise toutes les informations concernant les utilisateurs et leurs droits ;
- l'intégration harmonieuse au sein d'une infrastructure de gestion des clés PKI (Public Key Infrastructure) ;
- une unification des supports de communication (mails, forums, lidies, fax, SMS).

## II. Une solution avantageuse

### 1. Maitrise du TCO

Plusieurs problèmes ont été réglés pour assurer un coût réduit :

- l'adéquation aux besoins a été garantie par une évolutivité ;
- la montée en charge a donné lieu à une mise à l'échelle ;
- la spécificité et l'opacité ont demandé le respect de standards ;
- le coût de licence a été diminué grâce aux licences Open Source qui sont une garantie d'indépendance.

### 2. Points forts d'IDX-MU

Les avantages de la messagerie unifiée émanent directement de l'univers de l'Open Source ou des normes.

#### a. Les avantages liés à l'Open Source

##### - **Robustesse et fiabilité**

Cette solution peut tourner sur les plus gros serveurs de messagerie sur Internet.

##### - **Pérennité**

Contrairement aux solutions propriétaires, un code Open Source ne se perd jamais.

##### - **Sécurité**

Le code est facilement accessible, il est donc simple de détecter des failles. La correction est fournie en temps réel.

#### b. Les avantages liés aux normes

Ce produit s'appuie sur des normes, sur des protocoles comme SMTP qui permet le transfert de données, POP3 qui assure la récupération du courrier ou IMAP4 qui joue le même rôle mais en intégrant une dimension de centralisation et de sécurité supplémentaire. Il s'appuie également sur LDAP, une solution universitaire reprise par Microsoft et Netscape et qui évolue en Open Source sous la version Open LDAP. Enfin, SSL permet de crypter toutes les connexions d'échanges informatiques.

#### c. La sécurité

Nous intégrons également la solution IDX-PKI qui fut parmi les premières infrastructures à clé publique en Open Source. Elle offre la possibilité de chiffrer et signer les *mails*. : l'utilisateur peut garantir l'authenticité et l'intégrité du message.

## III. La démarche IDEALX

Nous accompagnons notre client pendant l'intégralité du processus. Nous offrons également un support technique de second niveau pour les administrateurs système.

**Bernard LANG**

Cette solution est-elle déjà installée ?

**Serdjane TIJANIC**

Cette solution est utilisée en interne chez IDEALX et a été présentée à Linux Expo. Quelques projets sont en cours chez des acteurs majeurs qui demandent un certain niveau de confidentialité.

La solution utilise des briques qui ont fait leurs preuves, notamment dans le cadre de la messagerie Yahoo !.

**De la salle**

Sous quel système d'exploitation fonctionne votre solution ?

**Serdjane TIJANIC**

IDX-MU fonctionne sous Linux et sous toutes les familles BSD (FreeBSD) qui assurent une bonne montée en charge et une bonne exploitation du matériel. Nous nous appuyons aussi sur OpenBSD qui s'attache plus à la sécurité.

**De la salle**

Vous évoquez un bloc-notes, faites-vous de la réplication, avez-vous un moteur de recherche ?

**Serdjane TIJANIC**

Effectivement, le moteur de recherche est possible du fait de l'utilisation de briques.

**De la salle**

Faites vous de la réplication de *mails* ?

**Serdjane TIJANIC**

Toutes ces fonctionnalités sont intrinsèques. Le concept de migration est en cours de réalisation pour un client. Il permet de récupérer en continu tous les messages dans plusieurs boîtes à lettres distantes et de les rapatrier vers une messagerie unique.

## **De la salle**

Sendmail ne rend-il pas déjà ce service ?

### **Serdjane TIJANIC**

Il est possible d'intégrer Sendmail dans cette solution. Nous sommes plus orientés vers une utilisation de Postfix. Sendmail a certes contribué à la normalisation du protocole SMTP, mais il paye désormais le prix de son statut de pionnier. Le coût d'un *fork* avec Sendmail est plus élevé qu'avec Postfix. L'avantage de Qmail et de Postfix est leur architecture modulaire : dans un système de messagerie, de nombreuses fonctionnalités doivent être traitées simultanément et doivent donc être réparties pour être plus légères.

### **Bernard LANG**

Est-il possible d'installer votre solution si l'on possède des applications Unix ou propriétaires ?

### **Serdjane TIJANIC**

Effectivement, toutefois notre objectif est de réduire les coûts et le déploiement d'une solution de messagerie sur une station Sun revient très cher.

### **Bernard LANG**

Quelle est l'alternative sur une plate-forme Sun ?

### **Serdjane TIJANIC**

Je connais l'exemple d'une société qui utilisait un annuaire Notes en collaboration avec Sendmail.

### **Bernard LANG**

Vous êtes en concurrence avec IBM.

### **Serdjane TIJANIC**

Une technologie Open Source n'a pas de concurrence. Nous ne cherchons que des solutions.

### **Bernard LANG**

Quelle est la compatibilité avec les autres systèmes ?

**Serdjane TIJANIC**

Dès lors que nous aboutissons sur une norme standard, la communication est possible avec tous les systèmes.

**De la salle**

Si une entreprise utilise Sendmail, est-elle obligée de migrer ?

**Serdjane TIJANIC**

Je n'ai pas indiqué une seule version de logiciel ou de brique Open Source. Nous procédons à des recommandations, nous préconisons Open LDAP ou Postfix, amis si nos clients disposent d'une expérience ou d'une compétence Sendmail, nous ne pouvons pas les obliger à migrer.

**De la salle**

La gestion de l'annuaire est-elle assurée par Open LDAP ?

**Serdjane TIJANIC**

La version libre de LDAP permet une administration décentralisée avec une délégation.

**De la salle**

Quelle est la taille des annuaires ?

**Serdjane TIJANIC**

Mon expérience se fonde sur un annuaire de 10 000 utilisateurs sur FreeBSD fonctionnant à partir d'un ordinateur personnel équipé d'un processeur Intel Pentium II cadencé à 300 Mhz avec une mémoire vive (RAM) de 128 Mo. Cet exemple se trouve sur le site pour enfants sssplash.fr.

# Linux et architecture fine (thin clients, thin servers)

**Jean-Pierre LAISNE**  
Président, Linbox

## **I. Qu'est-ce que le « fin » ?**

### **1. Un besoin**

- Les utilisateurs ne sont pas forcément des administrateurs système. L'idéal de Microsoft qui veut faire de tout utilisateur un administrateur système n'est pas possible.
- Chaque dépannage par un technicien coûte cher.
- Chaque site a besoin d'un traitement personnalisé.
- Les coûts, en général, sont extrêmement élevés. Le déploiement, c'est-à-dire l'installation et le paramétrage, d'une application SAP payée 10 millions de francs coûte dix fois plus cher. Le Gartner Group a inventé le concept de coût total de possession (Total Cost of Ownership, TCO) pour indiquer que le coût véritable d'un équipement informatique ne se révèle pas à l'achat mais à l'utilisation. Si nous prenons pour base l'acquisition d'un matériel amorti sur trois ans, le coût d'achat représente 21 % du TCO, la formation des utilisateurs représente 50 %, l'administration comptable des licences, 14 % et le support technique 15 %.

Nous déduisons de cette démonstration que la complexité de l'informatique est coûteuse. Pour réduire ce poste de dépense, il faut simplifier la vie de l'utilisateur final qui n'est pas un informaticien. Les utilisateurs ont besoin de configurations simples à maîtriser et à personnaliser.

### **2. Retour vers le futur**

Il est courant de parler d'applications Internet, d'ASP, de post-PC, de PDA pourtant, si le taux de pénétration de l'informatique dans les foyers américains atteint 60 %, il ne faut pas oublier que le reste de la population n'a jamais touché un ordinateur. Le marché n'est donc pas saturé pour de nouvelles interfaces, de nouveaux modes de connexion ou de traitement de l'information.

Nous nous trouvons dans un paradoxe : le Web et les applications Internet en général imposent la présence de gros serveurs similaires aux Mainframes des années 1980. Nous assistons donc à un retour vers une informatique centralisée qui fut un temps abandonnée avec l'avènement de l'ordinateur personnel.

L'industrie informatique est conçue à partir de composants tant logiciels que matériels. Une infrastructure se dessine comme une maison à partir de briques.

## II. Les acteurs du « fin »

### 1. Les serveurs fins

Est fin tout appareil connecté sur le réseau, et qui nécessite un minimum d'installation et d'administration. et dont tous les composants sont administrables à distance. Cette tendance est apparue au siècle dernier, dans les années 1990. Des acteurs sont apparus et se sont positionnés à la fois sur le client et sur le serveur. Cette définition du « fin » convient à ces deux aspects.

Les serveurs travaillent tous sur le même réseau, si l'un d'entre eux tombe en panne, les autres continuent à travailler. La logique qui prédomine ici impose une séparation des éléments pour assurer la solidité de l'ensemble.

Cobalt est l'acteur majeur des années 1990. Cette société a été rachetée par Sun pour 2 milliards de dollars en échange d'actions. Cobalt fut le précurseur sur ses solutions emballées. Le Cube et l'Oracle sont des serveurs de communication qui supportent des fonctions standard et des serveurs Web, *mail*, similaire à NT comme Samba qui assure aussi un filtrage et la sécurité.

A la suite de l'exemple de Cobalt, plusieurs pionniers se sont lancés dans les applications serveur et Internet : Encanton, Netwinder, Whistle ou Watchguard. Ils ont été rejoints ou rachetés pour certains par de grands acteurs traditionnels comme IBM ou Dell. Tous ces acteurs travaillent sous Linux. Certaines applications serveur servent à agrandir les capacités de stockage sans procéder à des opérations d'administration compliquées (Raptor ou Network Appliance). Certains prévoient l'installation du serveur Web à partir d'un simple disque compact (E-Smith). Demolinux est un outil de prosélytisme Linux : il permet de s'essayer à Linux sans l'installer. Le CD boote en mémoire et sert de disque. Linbox a signé un contrat avec demolinux pour, à partir de cette technologie, faire des *firewalls* et des serveurs qui ne s'installent pas. Cette solution est permise de cumuler une sécurité irréprochable et une ergonomie rare.

Tivo est un magnétoscope numérique. Il présente un modèle d'affaires intéressant : il couple une innovation technologique à un service d'abonnement.

### 2. Les clients fins

Les Winterm de Wyse ou de Igel utilisent Linux. VNC est un logiciel libre développé au départ par Olivetti et Oracle et qui est aujourd'hui repris par AT&T pour le *Webphone*.

Les applications Internet connaissent des difficultés pour décoller.

Ensuite, viennent les « ardoises » qui permettent aux métiers de logistique et d'inventaire de posséder une connexion vers leur système d'intervention.

### 3. Linbox

#### a. Notre offre

Linbox fournit des systèmes réseaux fondés sur une configuration qui allie les aspects clients et serveurs fins. Nous installons automatiquement un serveur qui contient toute l'intelligence du système d'information

et sur lequel on connecte *via* le réseau des stations clientes qui récupèrent un OS complet. Ce système d'exploitation permet de faire tourner un environnement Linux ou Windows en le sécurisant.

L'architecture est simple : il s'agit d'un réseau *Plug & Forget*.

#### *b. Nos clients*

Nous venons de réaliser le déploiement de 850 serveurs pour la Direction générale des impôts.

Nous travaillons également avec Hill House Hammond, un courtier en assurances qui possède 250 agences et 4 000 utilisateurs. Grâce à notre technologie Linux, chaque site est automatiquement installé en moins de quinze minutes.

Notre architecture est particulièrement bien adaptée au milieu éducatif et scolaire. Il s'agit d'une salle de formation toujours « propre ». Le redémarrage du système garantit son intégrité.

#### *c. Comment réduire le TCO ?*

- En automatisant un maximum de services d'administration, on évite toute intervention humaine (*end user operations*).
- La centralisation sur le serveur permet cette réduction des *end user operations*.
- L'architecture *Plug & Forget* automatise le dépannage à distance.
- La sécurité et le contrôle du système d'information sont améliorés.

### **III. La croissance du marché**

Les études IDC de 2000 montrent un part de marché de 27 % de Linux. Ces architectures fines sont le plus souvent gérées par des logiciels libres.

#### **1. Pourquoi Linux ?**

- Linux assure la robustesse, la fiabilité et l'évolutivité.
- C'est un logiciel multi plate-forme.
- Les logiciels libres et l'Internet se sont développés en étroite collaboration.
- Le coût des licences est nul.
- Il n'existe pas d'héritage informatique à supporter.
- Linux est commercialisé par de nombreux développeurs.
- Son marché explose.

- Il est très utilisé par les laboratoires de recherche.

## 2. Perspectives

Nous nous orientons vers la fin de l'ordinateur personnel et vers un outil communicant (*social computing*).

L'architecture fine essaie de rapprocher l'information de l'utilisateur final.

Les supports sont variés : Web, assistant de données personnel (PDA), téléphones mobiles grâce aux protocoles d'accès sans fil (WAP), kiosques, points de vente, guichets automatiques, bureau virtuel.

Le marché de masse est à portée de main : très petites entreprises, Small Office, Home Office (SOHO), l'éducation, les familles et les tribus.

La simplicité est un gage de solidité. Tout investissement doit tenir compte des barrières à l'entrée et à la sortie. Les logiciels libres veulent satisfaire au maximum les utilisateurs : tout développement doit être utile à la communauté.

J'étais récemment en Grande-Bretagne où Microsoft est fortement implanté du fait d'accords avec le gouvernement et au cours de mes entretiens des consultants m'ont avoué penser que Linux s'y développera sous deux ans.

### **Bernard LANG**

Nous sommes confrontés à un nouveau modèle économique et technique qui a été permis par l'arrivée des logiciels libres. Le contrôle des logiciels permet de développer des solutions personnalisées.

### **Jean-Pierre LAISNE**

La tendance de l'industrie est à la personnalisation de masse. Dans le secteur du textile comme dans celui de l'informatique, on affine l'offre aux besoins des utilisateurs. Nous nous retrouvons dans le paradoxe de « retour vers le futur » avec les Mainframes.

### **De la salle**

Le back office tourne-t-il ?

### **Jean-Pierre LAISNE**

Bien sûr ! nous avons un environnement dans lequel tournent les applications Windows en attendant de les supprimer. Ainsi nous respectons la liberté de choix de l'utilisateur qui peut migrer pacifiquement d'un environnement vers un autre. Seules les applications spécifiques ne tournent pas.

### **De la salle**

Quel va être le coût pour une entreprise de 1 000 utilisateurs dont 400 ont besoin d'un logiciel Excel ? De quel type de licence d'agit-il ?

### **Jean-Pierre LAISNE**

Je vais vous vendre 400 licences Windows. Je facilite et sécurise l'usage mais je n'interviens pas dans la politique de délivrance des licences de Microsoft.

### **Bernard LANG**

A la différence des solutions propriétaire qu'il faut changer à chaque nouvelle version, ces solutions une fois maîtrisées peuvent être conservées *ad vitam aeternam*.

### **Jean-Pierre LAISNE**

Un investissement informatique s'amortit en trois ou cinq ans, les logiciels libres permettent de ne pas s'y intéresser pendant ces cinq années.

### **De la salle**

Quel est le temps de réponse ? Combien de postes peut-on trouver derrière un serveur ?

### **Jean-Pierre LAISNE**

Les deux éléments dépendent de la taille du serveur. La distribution des systèmes est possible tout en conservant la centralisation.

# Zope : un exemple de serveur d'application

Stéphane FERNIGIER

Nuxeo

## I. Présentation générale

Zope est un serveur d'application et un système de gestion de contenu. Il s'agit d'un environnement de développement collaboratif qui vise à créer et à gérer des sites dynamique.

### 1. Les services fournis par Zope

Zope fournit une architecture à trois niveaux :

- la gestion de la persistance de données ;
- la gestion de la logique métier ;
- la présentation générale sous forme de pages HTML.

Comme tout serveur d'application, Zope traite la sécurité et le déploiement de composants (produits). Enfin, tout est géré depuis l'interface Web entièrement banalisée. On a pas besoin de déployer des logiciels spécifiques sur des postes clients.

### 2. Historique

Zope est fondé sur un logiciel créé en 1996 par une société américaine, Digital Creations. En 1998, un investisseur a insisté pour que le logiciel devienne libre et change de nom. Il devient Zope et se crée une communauté d'utilisateurs qui deviennent un marché. En deux ans, Zope est devenue une des plate-forme leader dans le domaine du logiciel libre en étroite relation avec le langage Python. D'ailleurs, l'équipe de développement du langage a été embauchée par Digital Creations.

L'aspect développement collaboratif communautaire est crucial, il repose sur :

- la discussion ouverte des spécifications ;
- l'intégration des composants développés à l'extérieur de Digital Creations.

### 3. Pourquoi utiliser Zope ?

- Il s'agit d'un logiciel libre qui respect de nombreux standards et qui est multiplate-forme.
- Zope se suffit à lui-même, il n'est pas nécessaire de posséder un serveur de données, un serveur LDAP, un serveur Web... Ensuite, il est possible de se connecter à d'autres sources de données extérieures (serveurs Web, LDAP, IMAP/POP).

- La stabilité est assurée, la montée en charge passe par une solution *clustering* qui permet de répliquer des bases de données Zope et d'avoir de nombreux serveurs qui répondent qui répondent.
- Zope est conçu pour publier des objets.
- Les postes de travail peuvent être banalisés.
- L'interface ressemble au navigateur de fichiers traditionnel et l'accès peut se faire depuis un protocole WebDAV avec un client comme Internet Explorer.
- Le travail en équipe est facilité par la séparation du contenu et de la présentation, par les versions et les « undo » qui sont illimités.

#### 4. Produits similaires

##### a. Produits propriétaire

On peut citer d'une part les serveurs d'application du type J2E et d'autre part des plates-formes de gestion de contenu. Ces produits sont relativement onéreux.

##### b. Logiciels libres

Enhydra est un serveur d'application de type J2E et on peut citer pour la gestion de contenu des produits comme Midgard.

Un certain nombre de portails média utilise Zope dont Multithématiques, des sites de logiciels libres, des sociétés des sites institutionnels dont le SIT du Bas-Rhin, des sites liés à l'enseignement.

Les prestations sont assurées par des sociétés comme Digital Creation. Nuxeo est une société récente, spécialisée sur Zope et qui fait de la formation, du support, de l'hébergement et du développement (intégration, personnalisation de produits existants ou développement de nouveaux produits).

## II. Présentation technique

Zope repose sur des composants, sur un serveur Web, sur un langage de *templates*, une base de données objets et un système de fixation d'objets.

### 1. Zope Publisher

La publication d'objets s'appuie sur la correspondance d'une suite d'objets dans les URL et dans la base de données.

## 2. Acquisition

Dans un site Web, les objets doivent acquérir un certain nombre de propriétés du répertoire qui les contient. Ce concept est orthogonal à celui d'héritage de classes.

## 3. Sécurité

Il existe un système de permission pour chacun des objets.

## 4. Z Templates

Le langage de *templates* s'appelle DTML avec une syntaxe simple qui permet d'insérer les valeurs dans une page, de faire des boucles, des conditions...

## 5. Z Catalog

Il permet d'indexer les objets Zope et de construire facilement un moteur de recherche sur un site.

## 6. Accès aux bases de données

Il se fait grâce à des *drivers* pour les principaux systèmes du marché. Les requêtes utilisent une syntaxe SQL permettant de faire abstraction du SGBDR utilisé.

## 7. Cryptage

Zope peut être crypté en Python, en Perl.

## 8. XML

Une classe d'objets Parsed XML Document peut être scriptée à l'aide de l'API de W3C.

# III. Les cas d'utilisation

## 1. Sites statiques

Pour les sites statiques, Zope apporte un certain nombre d'avantages : il est possible d'utiliser des interfaces purement Web pour éditer des documents ou d'utiliser des clients traditionnels FTP ou WebDAV. Le Z Catalog permet d'obtenir un moteur de recherche gratuitement et un système de permission est disponible pour déléguer le contrôle à des sous-éditeurs.

## 2. Publication de nouvelles et de forums

Zope permet de publier des nouvelles et des forums grâce à des produits Quickshot.

## 3. Produits de Groupware

Il est compatible avec des produits de Groupware : Wordpilot, GUM/GUD/Zorganiser, Amphora/Amphora Light.

## 4. Gestion de connaissance

Le système Yihaw crée des annuaires hiérarchiques de type Yahoo ! mais de façon plus riche avec des foires, des questions, des nouvelles...

## 5. Workflow

### Jean-Paul SMETS, DRIRE Lorraine

La procédure de demande de stage à la DRIRE Lorraine demandait un temps de traitement de deux à trois mois. L'introduction de Zope a permis de mettre en place un formulaire : la procédure ne demande pas plus de deux semaines. Des *mails* sont envoyés successivement aux différents responsables qui doivent motiver leur refus le cas échéant. Deux avantages se dégagent en matière de *workflow* : la transparence et la rapidité de la prise de décision accroissent la satisfaction des utilisateurs.

### Stéfane FERNIGIER

L'exemple du SIT du Bas-Rhin a pour ambition de fédérer 3 000 personnes à travers 40 administrations. Le serveur doit s'intégrer dans 40 systèmes d'exploitation différents et il est impossible d'imposer un bureau uniforme à 3 000 personnes. Zope a été particulièrement utile pour intégrer cette diversité. De plus le prototype a été développé par des non-spécialistes à partir de produits existants.

### Bernard LANG

Cette installation a été primée hier soir dans le cadre de net 2001.

### Jean-Paul SMETS

Je précise également que ce site est hébergé sur celui de la DRIRE Lorraine.

### De la salle

Comment catégorisez-vous l'information pour savoir à qui la distribuer ?

### **Jean-Paul SMETS**

Les utilisateurs ont des droits distincts et peuvent être regroupés par catégories et dans le système de gestion documentaire chaque document peut se voir attacher un ensemble de propriétés selon une norme. A chaque connexion, le système détermine les données accessibles pour l'utilisateur.

### **De la salle**

Est-il possible de coupler l'ensemble avec un système de capture d'informations et de classification automatique ?

### **Stéfane FERNIGIER**

Cette possibilité est en cours de développement. Des agents vont rechercher des informations sur le site et la classifient. Il est donc possible d'en faire un système global de gestion de la connaissance.

# La sécurité et les logiciels libres

Denis DUCAMP  
HSC

Pourquoi utiliser les logiciels libres en matière de sécurité ?

## - **Enjeu commercial**

Les logiciels libres présentent moins de défauts du fait de l'absence de pression commerciale sur les délais d'édition. Les développeurs peuvent à loisir peaufiner leur logiciel.

## - **Lisibilité du code Java**

Le code peut être relu par des experts et les corrections demandent moins de temps. Il n'est pas question de *patches*. Plusieurs versions de Bind évoluent en parallèle du point de vue de la sécurité : seule la dernière version connaît de nouvelles fonctionnalités mais la sécurité peut être mise à jour sans devoir changer de logiciel.

## - **Pérennité des logiciels**

Dès qu'un logiciel est intéressant et que les sources sont disponibles, il peut être continué indéfiniment même après la disparition du développeur initial. Il peut y avoir des ruptures (*splits*) le logiciel est alors poursuivi par deux personnes différentes sous des noms différents. Par exemple, NetBSD a donné lieu à une dispute entre ceux qui voulaient maintenir sa très grande portabilité et ceux qui voulaient renforcer la sécurité. Les premiers ont poursuivi NetBSD alors que les autres ont lancé le code OpenBSD.

## - **Qualité de la documentation**

La documentation est très fournie et bien faite, elle ne fait pas l'objet d'une exploitation commerciale : *Linux documentation projects*, les « how to », *FreeBSD Handbook*, *OpenBSD FAQ*...

Le code pouvant être relu, il sera bien écrit, lisible et modulaire. La qualité du code est souvent supérieure : IPFilter est plus efficace que les *firewalls* commerciaux. La veille technologique est inévitable : même si on utilise des logiciels libres, les sites permettent de se tenir au courant des nouvelles versions et des corrections.

Les logiciels libres sont tous dans Unix : cette expérience repose sur 30 ans d'attaques, de relecture, de correction et d'améliorations.

## - **Standards**

Sendmail et Bind ont contribué au développement des RFC.

Les standards sont le seul moyen d'assurer l'interopérabilité à tous les niveaux. Le standard IP assure la cohérence de l'Internet. Les piles IP des logiciels libres sont les meilleures. En filtrage, on trouve IPFilter, NetFilter, IP-Firewall et IPChains. Pour configurer un *firewall*, on peut utiliser PHP Firewall Generator, GFCC qui permettent de créer des règles pour IPChains et d'autres filtres.

Le relayage applicatif permet, dans le cadre d'un système vulnérable, de vérifier les URL et de les rejeter le cas échéant : Squid...

Les logiciels de test de vulnérabilité sont de qualité égale à celle de logiciels commerciaux : Nessus, Whisker, Nmap...

La détection des intrusions est une tâche très difficile et complexe assurée par Snort, Shadow (détection des scans lents), Bro...

Le chiffrement peut se faire au niveau du réseau ou de l'application, mais il ne fait que protéger la confidentialité, il ne sécurise pas.

En conclusion, les logiciels libres offrent :

- une meilleure lisibilité des problèmes ;
- une plus grande autonomie ;
- une réponse à des besoins spécifiques ;
- de la transparence

### **Bernard LANG**

L'Agence pour la sécurité nationale américaine (NSA) propose désormais une version sécurisée de Linux.

### **Stéfane FERNIGIER**

Dire qu'un système est sécurisé peut avoir deux sens : il peut être dépourvu de *bugs* ou il peut suivre certaines recommandations. La NSA a rajouté des fonctionnalités de sécurité à Linux.

# **Les logiciels libres dans les collectivités territoriales**

**Philippe ALLART**  
Communauté urbaine de Lille, Illico

L'expérience que je vais exposer concerne la région Nord-Pas-de-Calais et la communauté urbaine de Lille.

## **I. Le choix stratégique des logiciels**

### **1. L'expérience de la communauté urbaine de Lille**

Depuis 1998, la communauté urbaine utilise des logiciels libres pour sa messagerie en remplacement de solutions propriétaire. L'initialisation a été très rapide : en 3 jours, les installateurs ont créé 3 000 comptes à l'aide d'un script. Ensuite, la formation s'est déroulée de façon relativement autonome à partir des « How to ». Aujourd'hui, nous enregistrons un million de connexions par mois et le système peut fonctionner sans être réinitialisé pendant 350 jours.

Nous avons même transformé des PC en terminaux X à partir d'une disquette.

Nous avons développé des applications cartographiques à partir de logiciels libres d'abord sur HP Unix puis sur Linux. Dès 1994, la compilation a été réalisée avec GCC qui nous a apporté une plate-forme de développement standard puisque GCC est disponible sur tous les Unix.

### **2. La problématique des collectivités**

Sur les quelque 36 000 communes de France, une trentaine seulement dépassent les 100 000 habitants. A partir de ce seuil, les villes sont capables d'acheter des logiciels propriétaires et commencer à influencer le marché. Trente communes ne peuvent pas créer un marché du logiciel et sont donc dépendants des éditeurs. Nous sommes donc amenés à faire nos propres développements pour répondre aux besoins des 250 métiers qui sont représentés dans les communes. D'une façon générale, les collectivités territoriales ont :

- des besoins décalés par rapport à l'offre du marché ;
- une maîtrise inégale de l'outil informatique ;
- une informatisation lente et pénible.

### **3. Nos développements**

Nous avons développé un outil de conception assistée par ordinateur (CAO) adapté aux besoins des cartographes à la différence des outils de dessin assisté par ordinateur comme Autocad. Les SIG du commerce sont plus adaptés à la gestion marketing de territoires plus vastes. Certains outils ont été

développés sous Windows avec Delphi. Dans tous les cas les sources que nous développons sont mises en Open Source. Il est possible d'ajouter très facilement des informations cartographiques sur des fichiers.

Nos développements ont été également portés sur le Web après avoir été réalisés à partir de logiciels trouvés sur Internet (Ghostscript, Tcl, CVS...).

## II. Les développements futurs

Nous sommes confrontés à trois choix possibles.

- Nous pouvons continuer d'assurer la maintenance des logiciels. Dans mon équipe, seulement une minorité des techniciens maîtrise Unix : le déploiement des applications sur le Web fait augmenter le coût de la maintenance. Il nous faut donc embaucher.
- Nous pouvons remplacer le développement par des logiciels commerciaux. Néanmoins, l'offre du marché n'est pas adaptée à nos besoins. Il nous faudra renoncer à certaines fonctionnalités et certainement faire chuter le taux de satisfaction des utilisateurs. De plus, cette solution comme la précédente est donc coûteuse.
- Nous pouvons partager les codes sources et trouver des partenaires. En augmentant le nombre d'utilisateurs, celui de développeurs potentiels croît aussi.

## III. L'argumentaire : convaincre les collectivités territoriales

Le Larousse donne de la technologie les définitions suivantes : « étude des outils, des machines et des techniques utilisées dans l'industrie » et « ensemble des savoirs et des pratiques fondés sur des principes scientifiques dans un domaine technique ». En fait, quand nous utilisons des outils que nous ne pouvons pas étudier ou dont nous ne pouvons pas connaître le fonctionnement interne, nous ne sommes pas dans le cadre d'une technologie. Or le logiciel libre nous inscrit dans un environnement où effectivement nous réalisons une étude des outils que nous utilisons et nous permet de maîtriser les pratiques. En conséquence, le logiciel est une technologie à part entière. Le choix des logiciels libres arrive avant la définition de l'appel d'offre, il constitue un choix stratégique. Ils forment une technologie à part entière et le choix doit être réalisé en amont de la rédaction du cahier des charges. La mise en concurrence ne peut concerner que des sociétés qui proposent la même technologie.

Nous pouvons, par cette démonstration, inciter les collectivités avec lesquelles nous souhaitons travailler à faire le choix des logiciels libres. A partir de ce moment-là, les sommes investies peuvent servir à maintenir un certain niveau de service et d'embauche plutôt qu'à payer des droits et des licences.

Dans le cadre de l'intercommunalité prévue par la loi Chevènement, la communauté urbaine récupérerait la compétence et la maîtrise sur les nouvelles technologies de l'information et de la communication (NTIC). Un chargé de mission a été nommé pour évaluer la partie connectique (câblage des bâtiments). La partie logiciel donne lieu à un projet de transfert avec d'autres collectivités que nous avons informées de nos développements et de la disponibilité des sources. Nous pensons transférer les ressources, en termes de logiciels et de données, de la communauté urbaine de Lille vers les communes qui pourront les appliquer dans le cadre de services rendus directement aux administrés. Nous espérons également harmoniser le

territoire en assurant une cohérence entre les plates-formes, maîtriser les coûts et confronter les points de vue des villes et de la communauté urbaine.

Le choix est effectivement plus stratégique qu'économique :

- la disponibilité d'une plate-forme libre a permis ce consensus technique préalable à la coopération ;
- la motivation des groupes de travail n'en est qu'accrue ;
- les projets peuvent être plus rapidement définis afin d'obtenir de meilleurs financements.

### **Bernard LANG**

Le travail coopératif est une idée très intéressante qui ne se limite pas uniquement aux collectivités territoriales. Les entreprises peuvent faire ce choix dans le cadre de logiciels qui ne revêtent pas une importance critique pour leur fonctionnement.